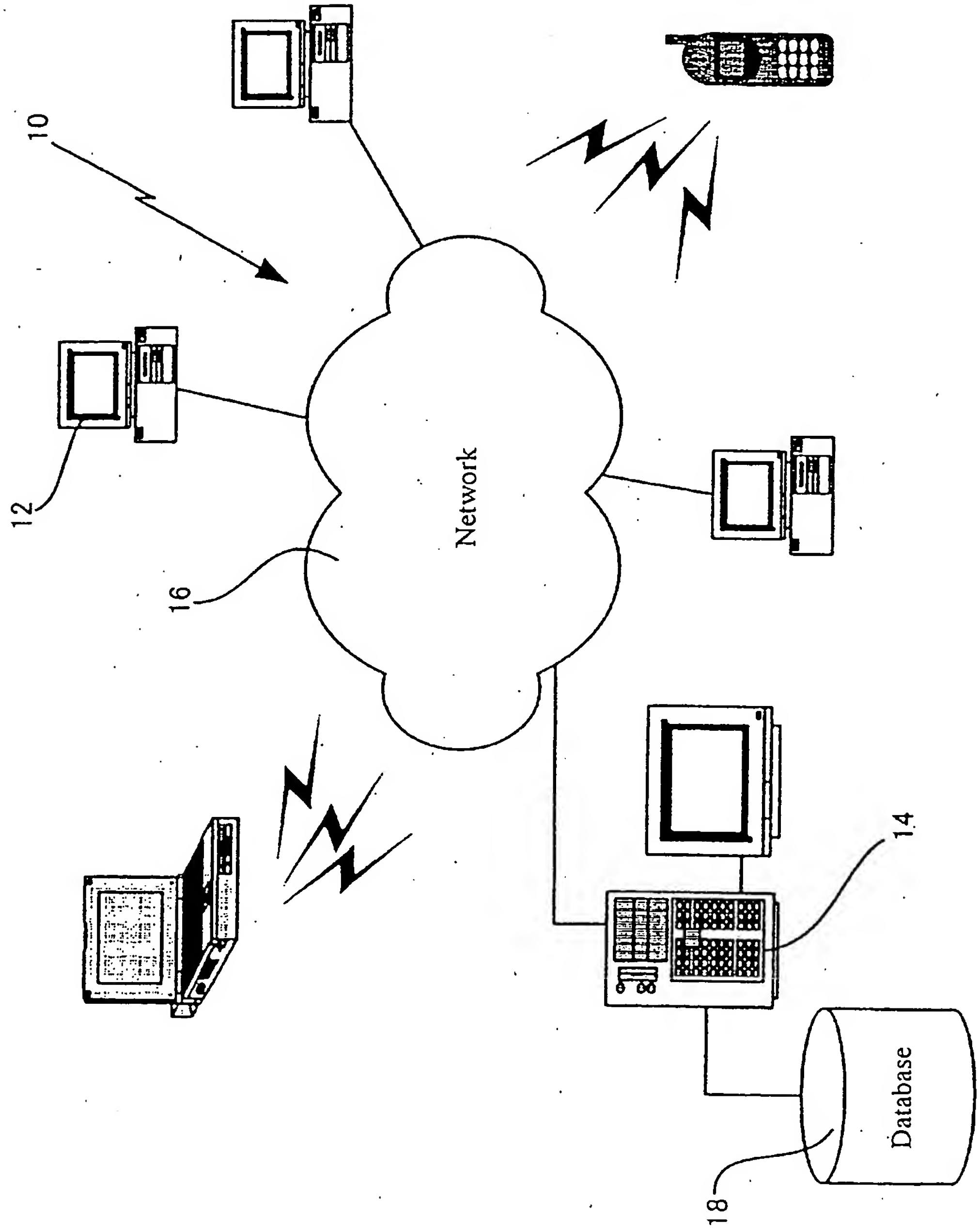
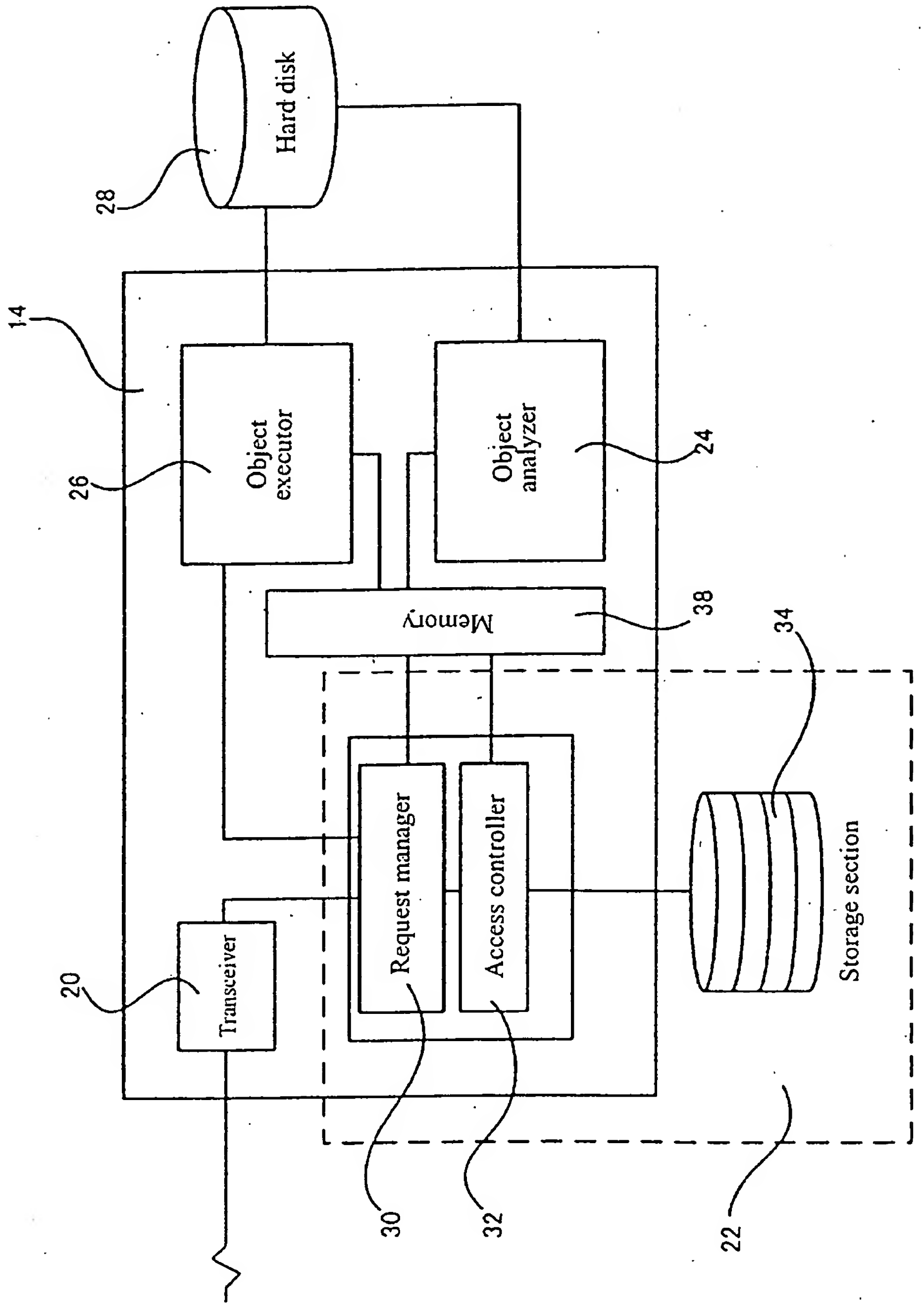
[Document type] Drawing

[Figure 1]

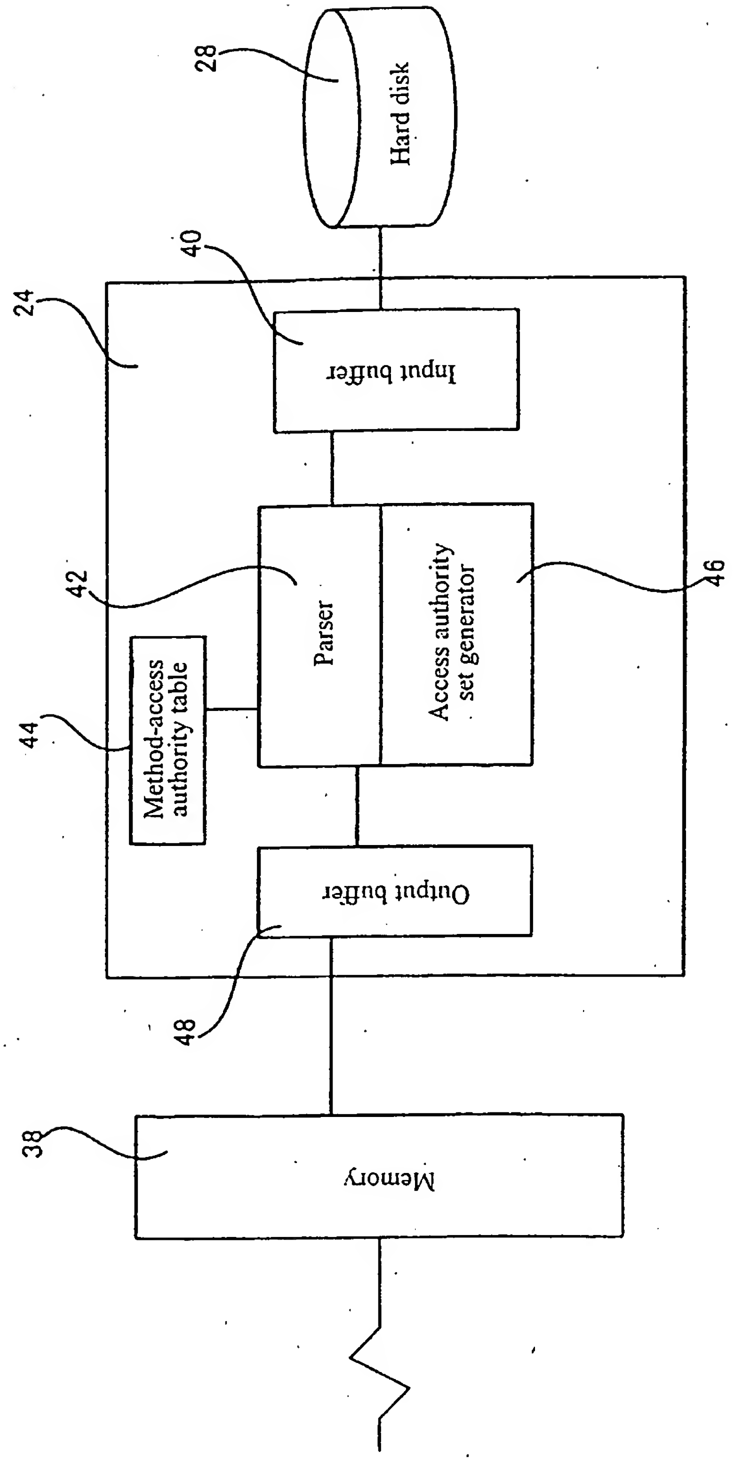
(1/23)



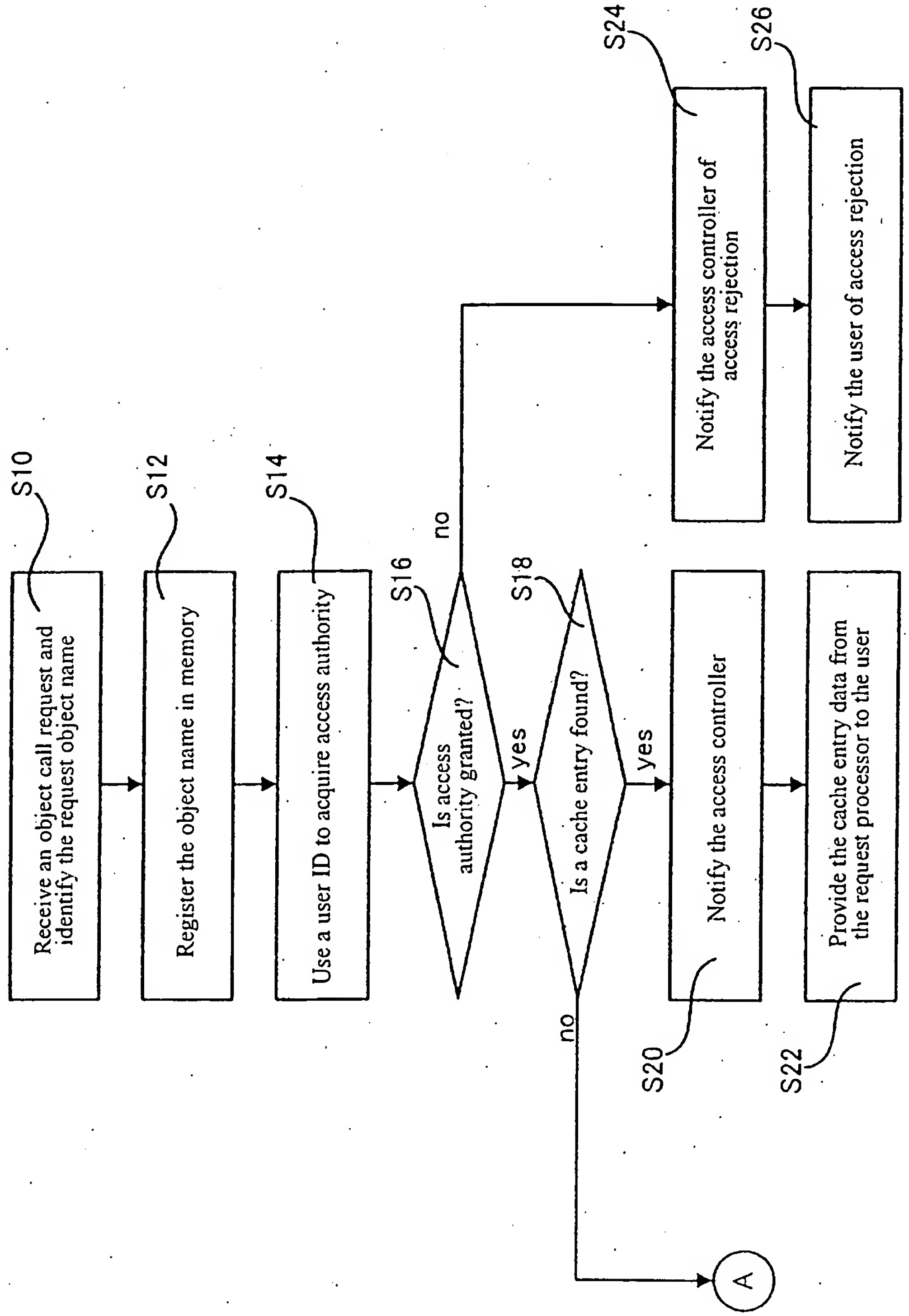
[Figure 2]


Hard disk
28
14
Object executor
26
Object analyzer
24
Memory
38
34
Request manager
Access controller
Transceiver
20
Storage section
30
32
22

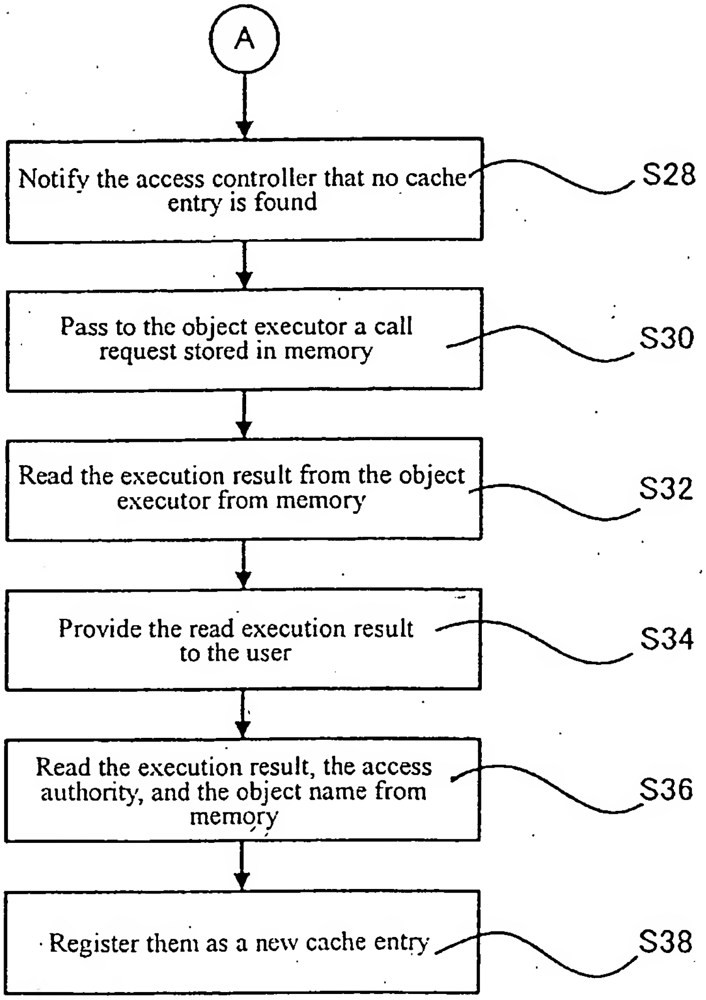
[Figure 3]


28
Hard disk
40
24
Input buffer
42
Parser
Access authority set generator
46
44
Method-access authority table
Output buffer
48
38
Memory

[Figure 4]


S10
Receive an object call request and identify the request object name
S12
Register the object name in memory
S14
Use a user ID to acquire access authority
S16
Is access authority granted?
no
yes
S18
Is a cache entry found?
no
yes
S20
Notify the access controller
S22
Provide the cache entry data from the request processor to the user
S24
Notify the access controller of access rejection
S26
Notify the user of access rejection
A

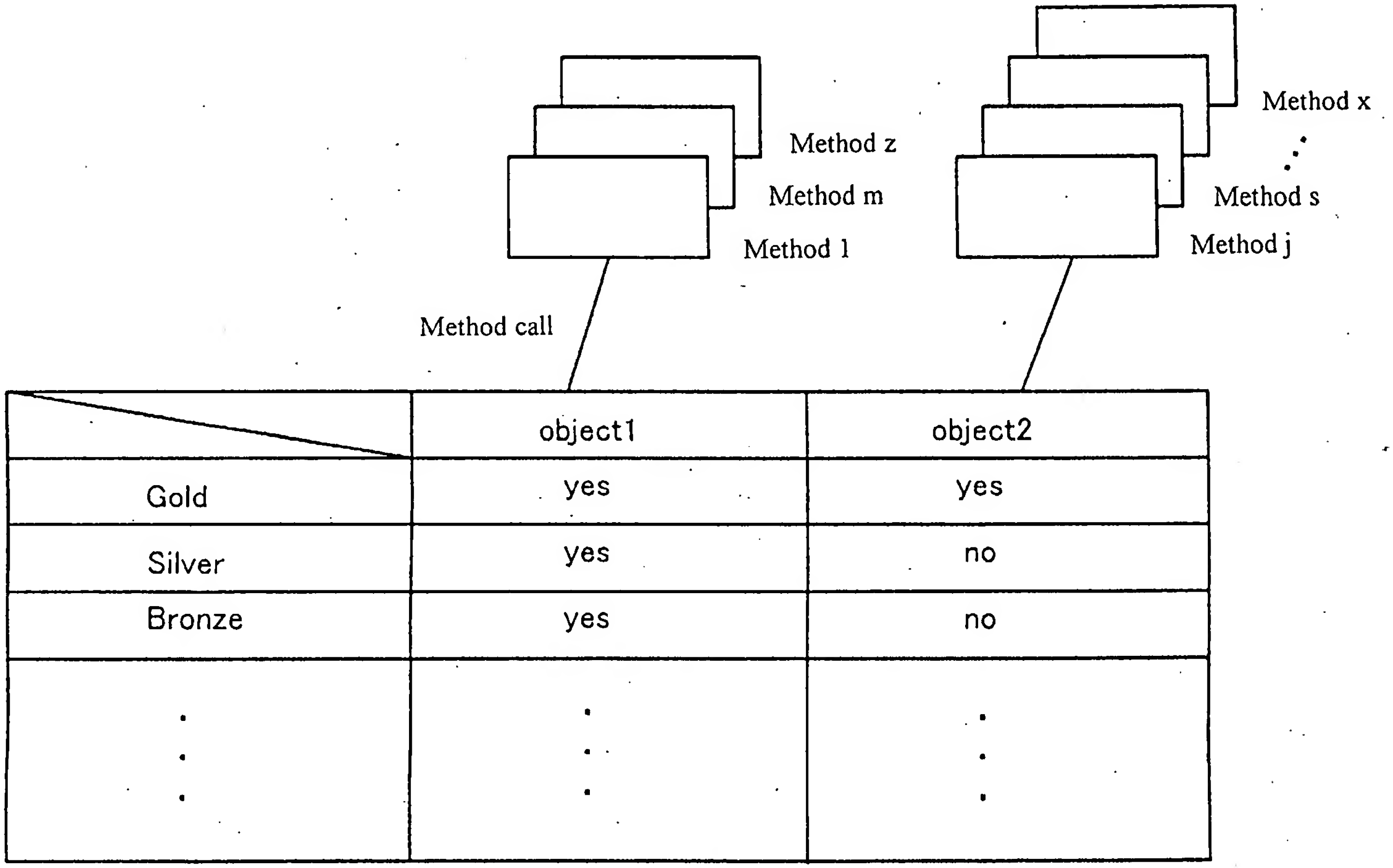
[Figure 5]


A
Notify the access controller that no cache entry is found
S28
Pass to the object executor a call request stored in memory
S30
Read the execution result from the object executor from memory
S32
Provide the read execution result to the user
S34
Read the execution result, the access authority, and the object name from memory
S36
Register them as a new cache entry
S38

[Figure 6]

User ID	Access authority	
usdxxhnm	gold	A, C, D, ···, X
usboojkn	gold	A, B, M, ···, X
bbiooean	bronz	B, C, M, ···, Y,Z
⋮	⋮	⋮
xjmbiwmka	silver	B, J, M, ···, X,Z
Ibm123trl	bronz	D, E, F, ···, O,...

[Figure 7]



	object1	object2
Gold	yes	yes
Silver	yes	no
Bronze	yes	no
⋮	⋮	⋮

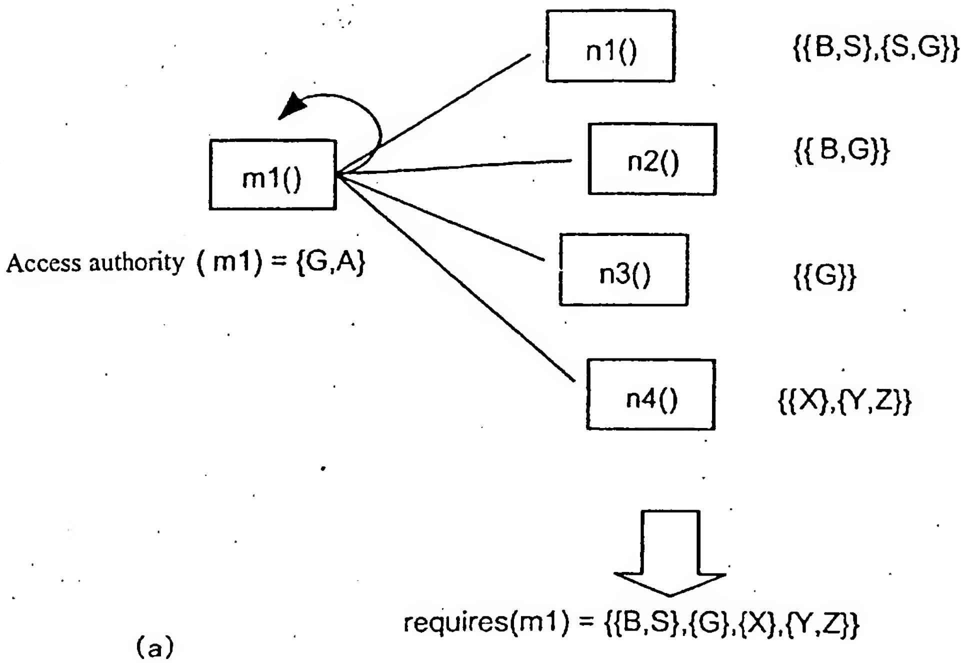
[Figure 8]

	Method 1	Method 2	...	Method i
Gold	yes	yes		yes
Silver	yes	no		yes
Bronze	no	no		no
A	yes	yes		yes
B	no	yes		no
C	no	no		no
⋮	⋮	⋮		

[Figure 9]

Object name	Access authority set	Execution result	
getweatherForecast ()	{(G,S),···,(A)}	Execution result 1	36b
getweatherForecast ()	{(B), (A)}	Execution result 2	
getDetailedInfo ()	{(G),···,(X,Y,Z)}	Execution result 3	36a
getStockRecommendation ()	{(G,B), (S), ···, (Y)}	Execution result 4	
getStockRecommendation ()	{(S), (S,B), ···, (Y)}	Execution result 5	36
⋮	⋮	⋮	

[Figure 10]


n1()
{{B,S},{S,G}}
n2()
{{B,G}}
m1()
n3()
{{G}}
Access authority (m1) = {G,A}
n4()
{{X},{Y,Z}}
requires(m1) = {{B,S},{G},{X},{Y,Z}}

(a)

$$requires(m) \cong \{permission(m)\} \oplus \overset{\oplus}{\bigcup_{i}} requires(n_i)$$

$$A \oplus B \cong \{r \in A \cup B \mid \neg(\exists s \in A \cup B; r \prec s)\}$$

$$r \prec s \Leftrightarrow r \cup s = s \wedge r \neq s$$

$$\overset{\oplus}{\bigcup_{i}} N_i \cong N_1 \oplus N_2 \oplus \ldots \oplus N_n$$

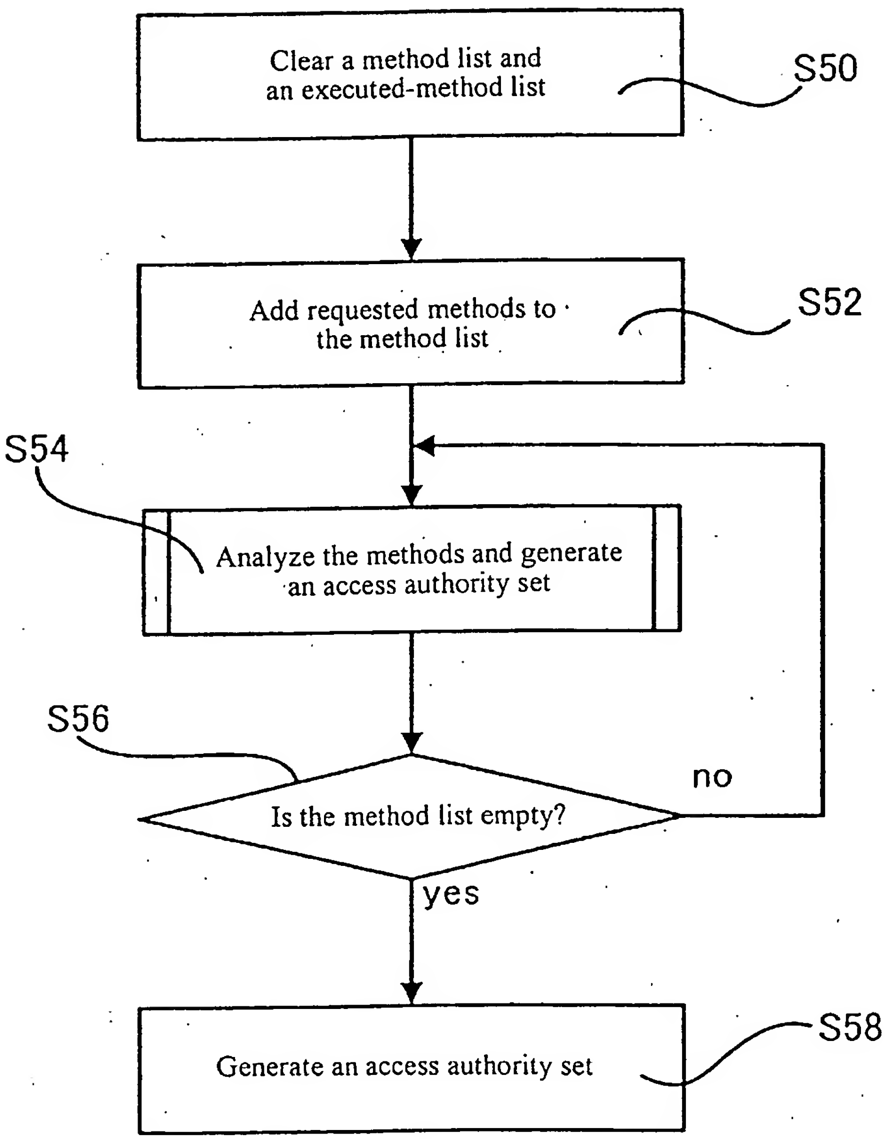
$$unknown(m) \Rightarrow permission(m) \cong \varnothing$$

$$safe(m) \Leftarrow \forall r \in requires(m); r \cap p \neq \varnothing$$

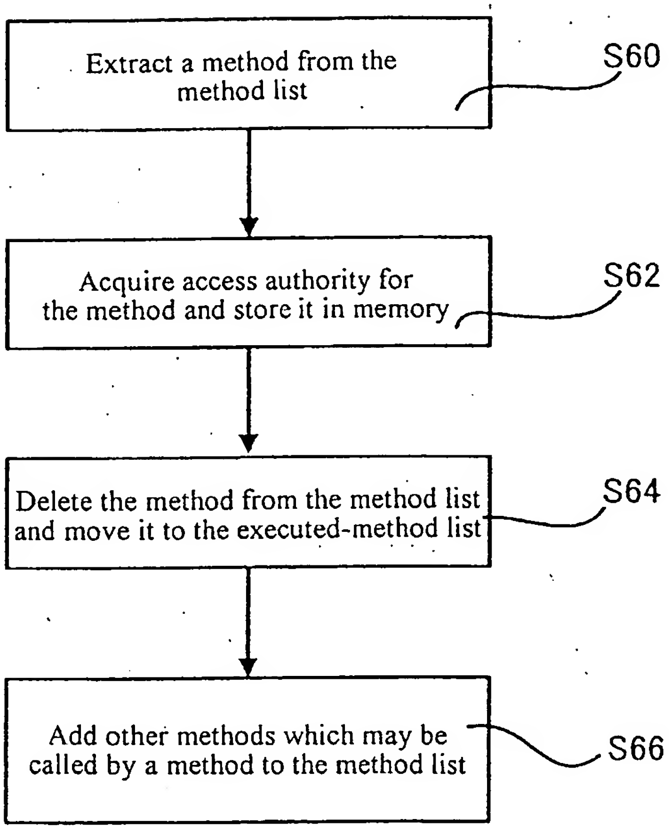
p := the roles - set of an accessing principal

(b)

[Figure 11]


Clear a method list and an executed-method list
S50
Add requested methods to the method list
S52
S54
Analyze the methods and generate an access authority set
S56
Is the method list empty?
no
yes
Generate an access authority set
S58

[Figure 12]


Extract a method from the method list
S60
Acquire access authority for the method and store it in memory
S62
Delete the method from the method list and move it to the executed-method list
S64
Add other methods which may be called by a method to the method list
S66

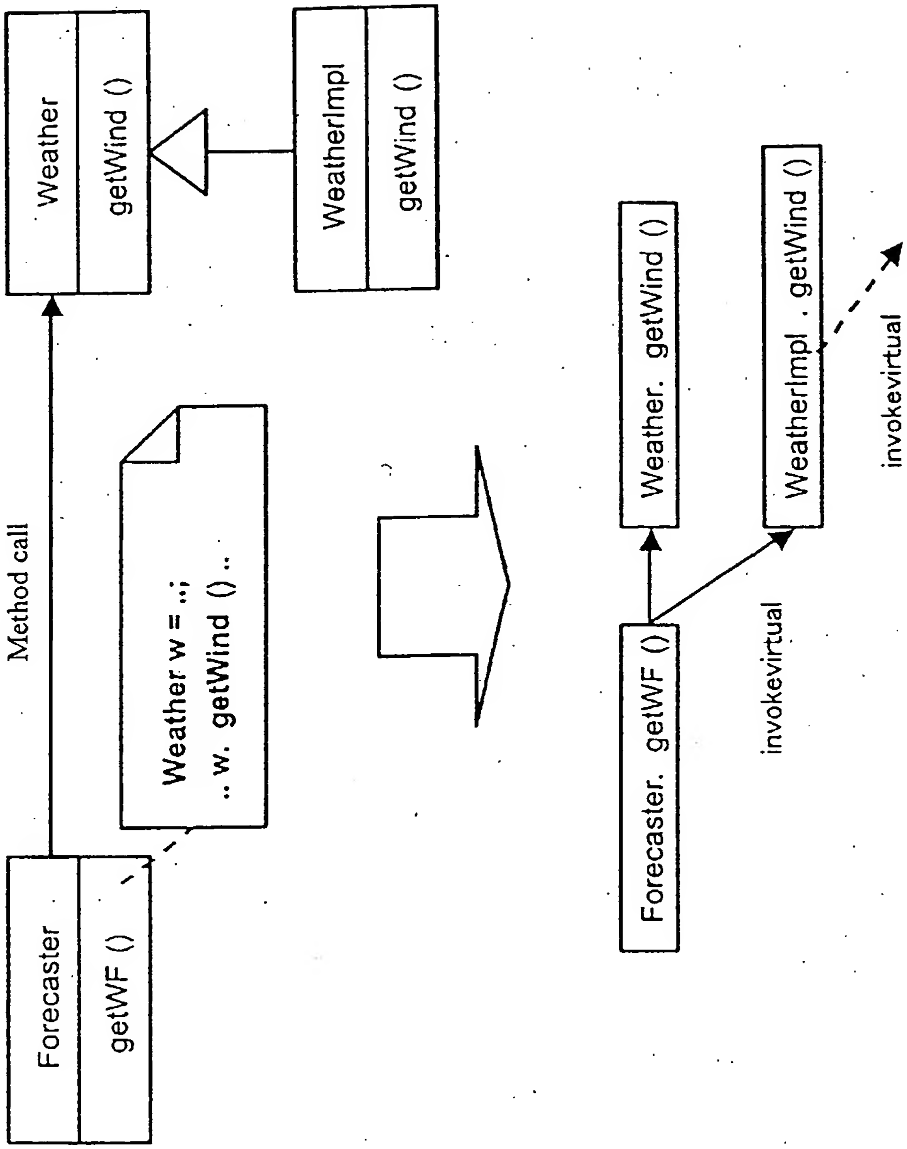
[Figure 13]

```
Set<Method> todo-list = {}
Set<Method> done-list = {}
Set<Set<Role>> result = {}

requiredRoles(method)

requiredRoles(Method m) {
    Set<Set<Role>> result = {}
    foreach Method n in depends(m) {
        if (n not in done-list) {
            add n to done-list
            add roles allowed for n to result (maybe by taking an optimized form)
            // may optimize result
            requiredRoles(n)
        }
    }
}
```

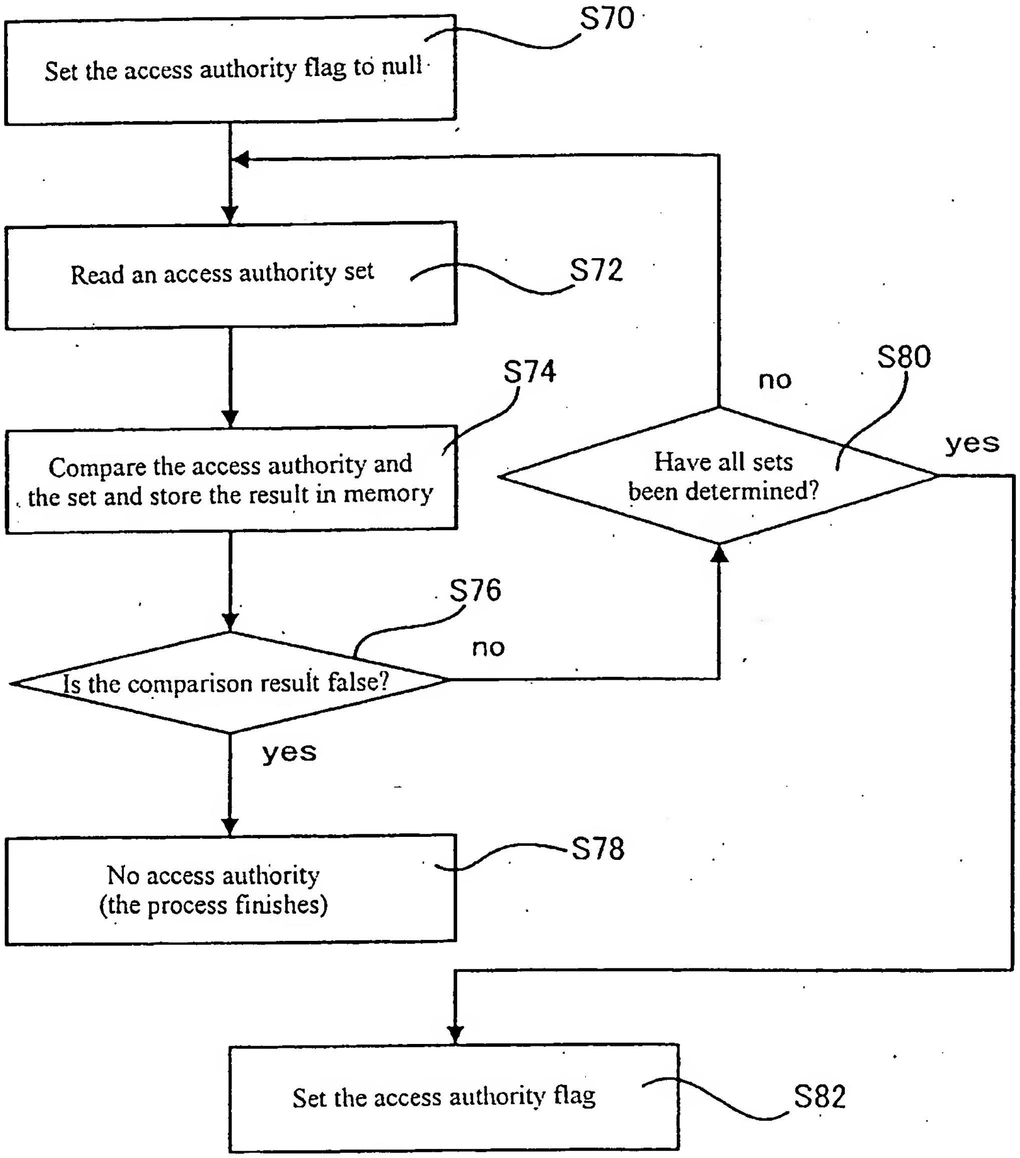
Set<Set<Role>> represents a set of a set of Role sets.

The depends() returns all of methods that may be directly invoked from the given method.

[Figure 14]


Weather
getWind ()
WeatherImpl
getWind ()
Method call
Weather w = ..;
.. w. getWind () ..
Forecaster
getWF ()
Forecaster. getWF ()
Weather. getWind ()
WeatherImpl . getWind ()
invokevirtual
invokevirtual

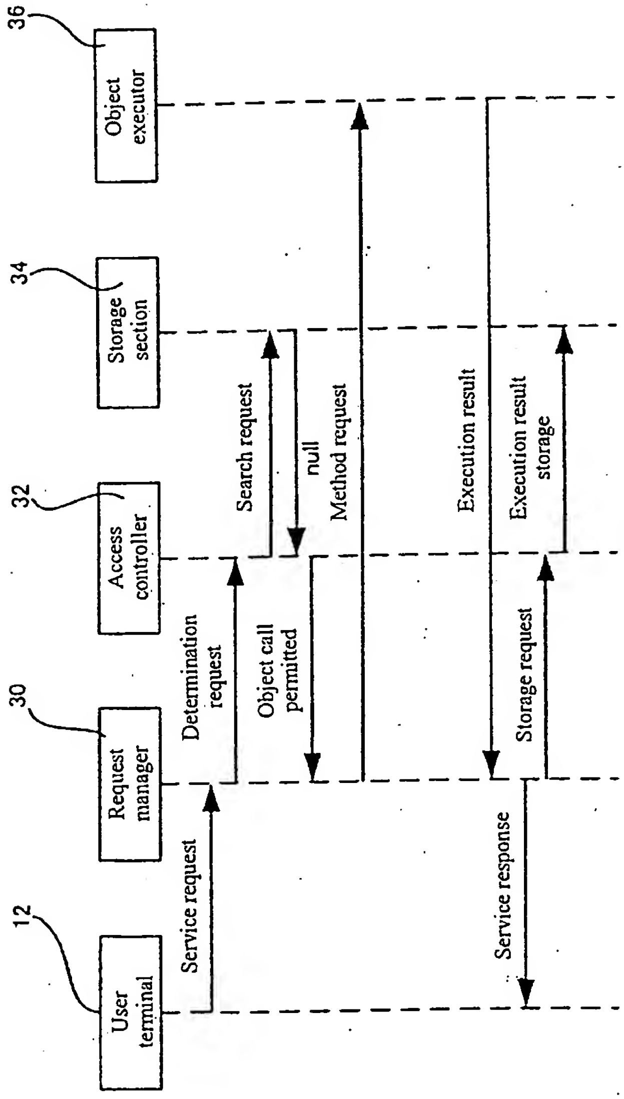
[Figure 15]


S70
Set the access authority flag to null
S72
Read an access authority set
S74
Compare the access authority and the set and store the result in memory
S76
Is the comparison result false?
no
yes
S78
No access authority
(the process finishes)
S80
Have all sets been determined?
no
yes
S82
Set the access authority flag

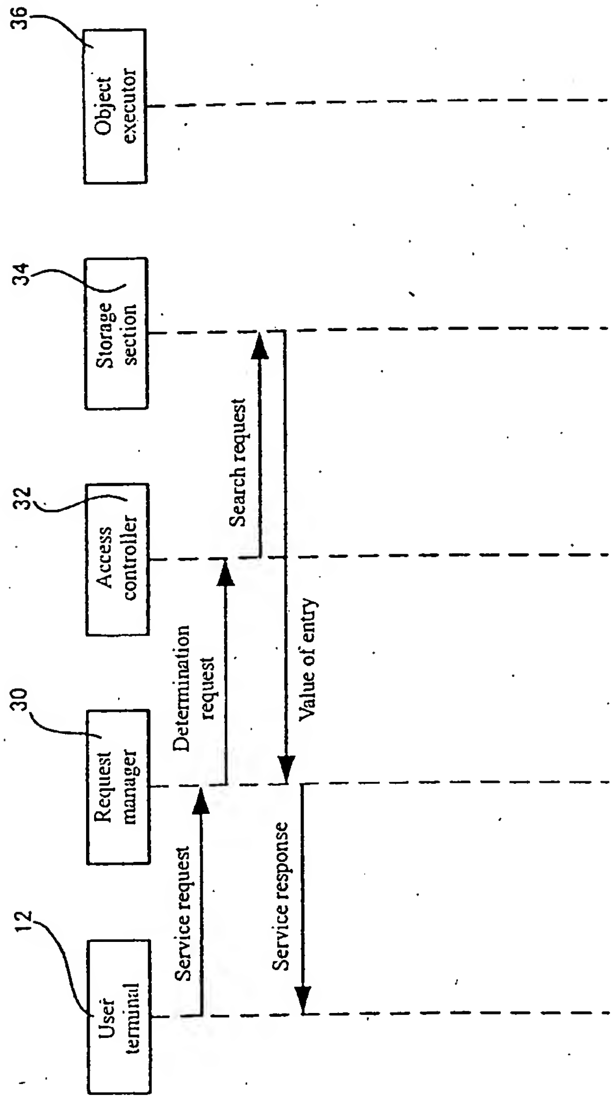
[Figure 16]

```
Principal p
Set<Set<Role>> roleCondition = { (A, B), (B, C), (X, Y, Z) }

if compliesWith(roleCondition) then success else failure

boolean compliesWith(Set<Set<Role>> roleCondition) {
      foreach Set<Role> allowedRoles in roleCondition {
          if not isInAnyOf(allowedRoles) return false
      }
      return true
}

boolean isInAnyOf(Set<Role> allowedRoles) {
      foreach Role role in allowedRoles {
        if p in role then return true
      }
      return false
}
```

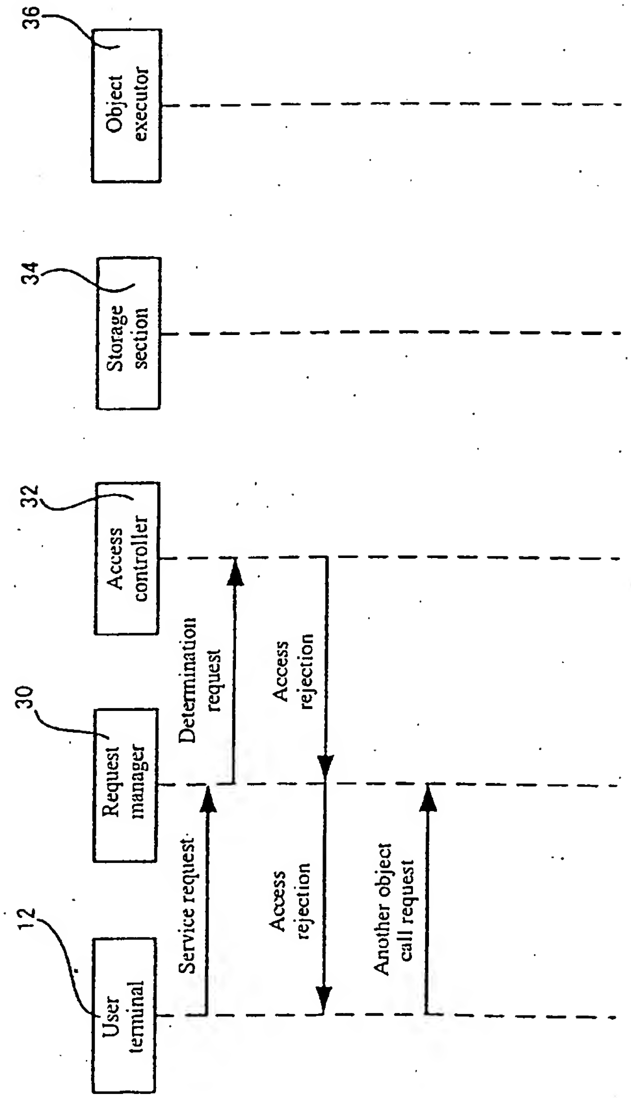
[Figure 17]


36
Object executor
34
Storage section
32
Access controller
30
Request manager
12
User terminal
Service request
Determination request
Object call permitted
Search request
null
Method request
Execution result
Execution result storage
Storage request
Service response

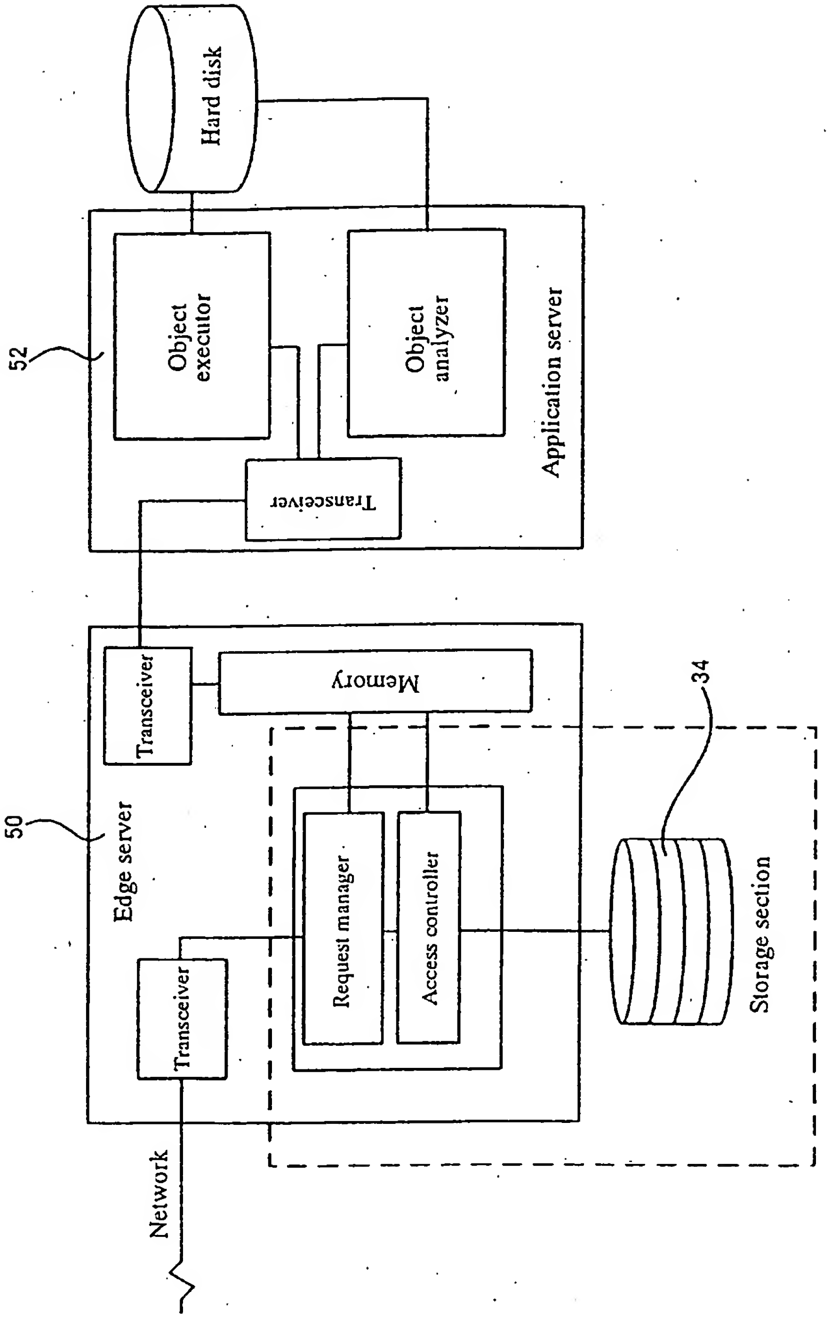
[Figure 18]


36
Object executor
34
Storage section
32
Access controller
30
Request manager
12
User terminal
Service request
Determination request
Search request
Value of entry
Service response

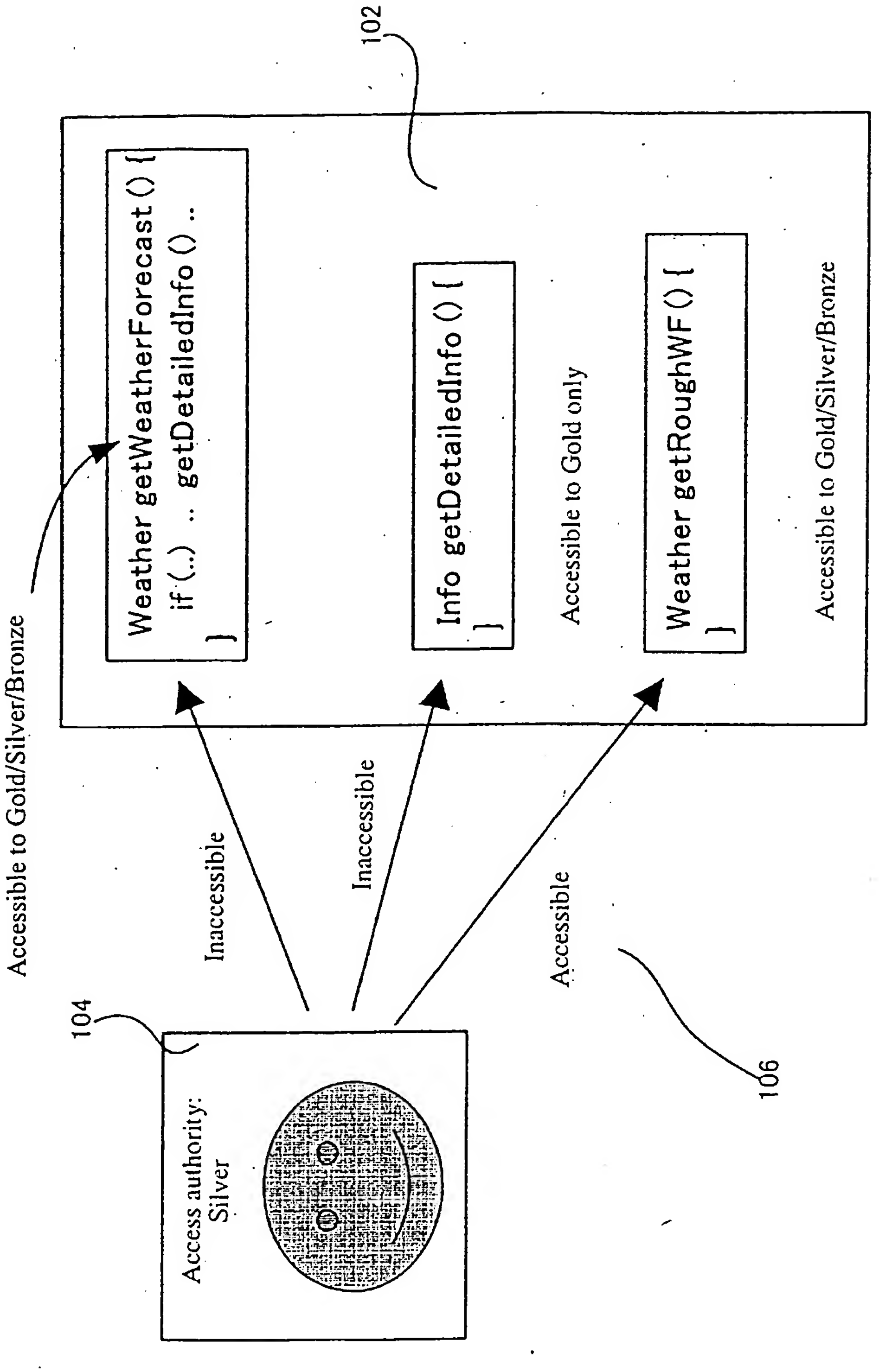
[Figure 19]


36
Object executor
34
Storage section
32
Access controller
30
Request manager
12
User terminal
Service request
Determination request
Access rejection
Access rejection
Another object call request

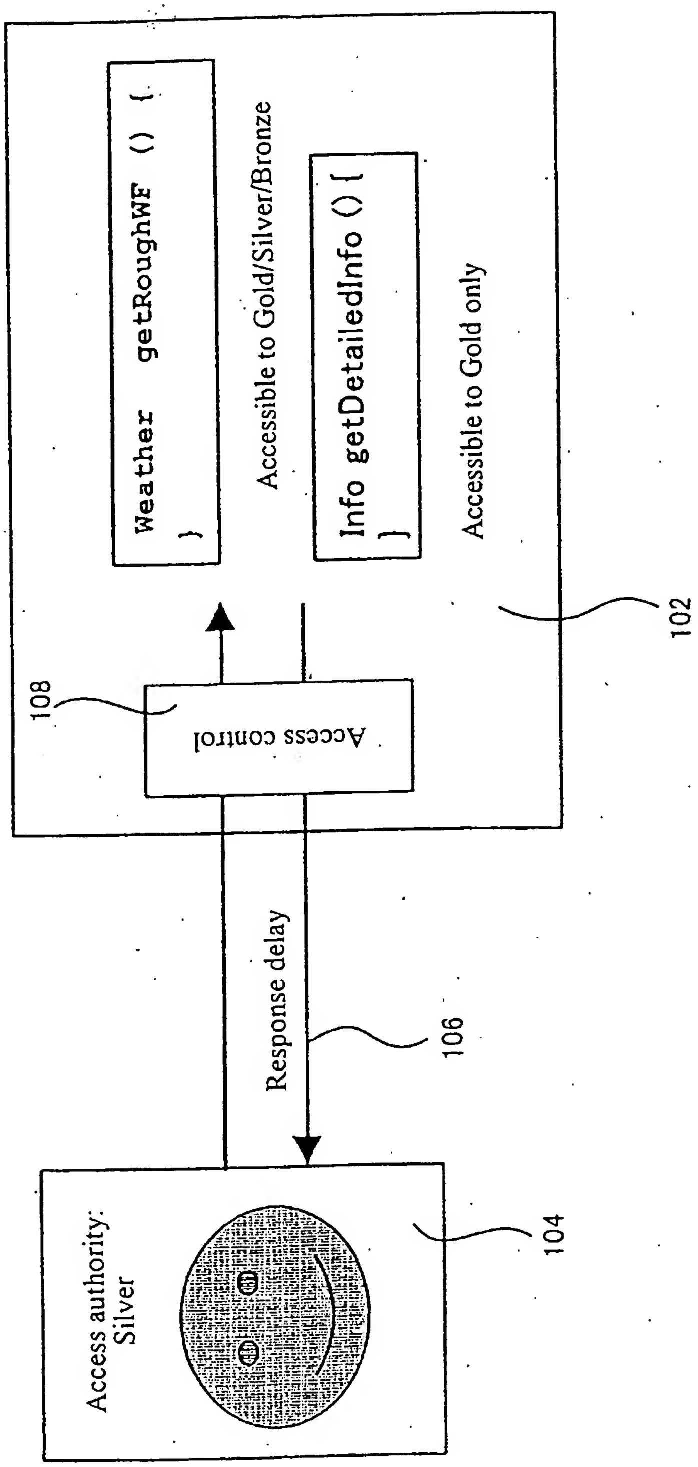
[Figure 20]


Hard disk
52
Object executor
Object analyzer
Transceiver
Application server
50
Edge server
Transceiver
Memory
34
Request manager
Access controller
Transceiver
Storage section
Network

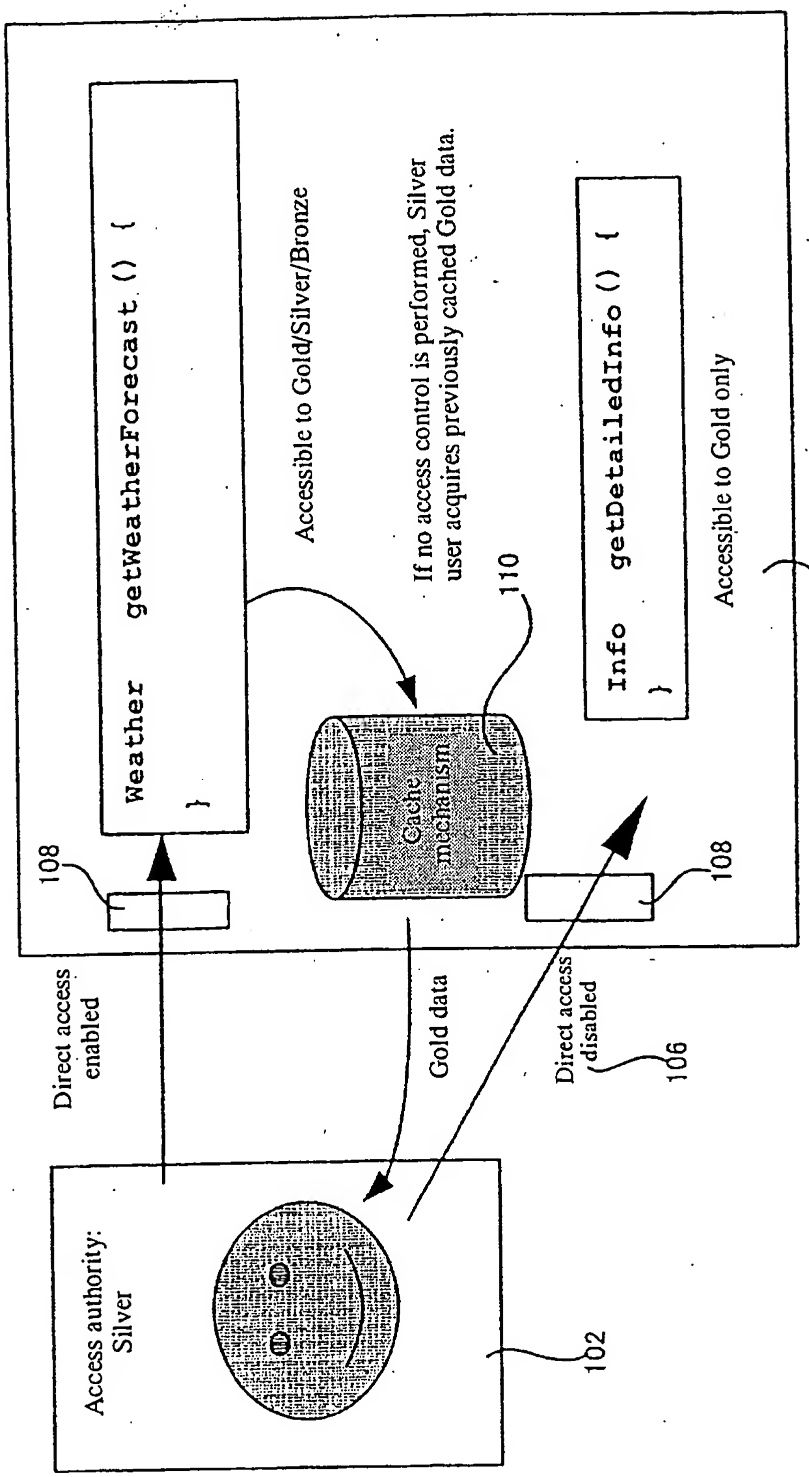
[Figure 21]


102
Weather getWeatherForecast () {
if (..) .. getDetailedInfo () ..
}
Info getDetailedInfo () {
}
Accessible to Gold only
Weather getRoughWF () {
}
Accessible to Gold/Silver/Bronze
Accessible to Gold/Silver/Bronze
Inaccessible
Inaccessible
Accessible
104
106
Access authority: Silver

[Figure 22]


Weather getRoughWF () {
}
Accessible to Gold/Silver/Bronze
Info getDetailedInfo () {
}
Accessible to Gold only
102
108
Access control
Response delay
106
Access authority:
Silver
104

[Figure 23]


Weather getWeatherForecast () {
}
Accessible to Gold/Silver/Bronze
If no access control is performed, Silver user acquires previously cached Gold data.
110
Info getDetailedInfo () {
}
Accessible to Gold only
104
Cache mechanism
108
108
Direct access enabled
Gold data
Direct access disabled
106
Access authority: Silver
102